GUI Based Tool for Watermarking Attacks

Jovan Spasik[1], Dimitar Taskovski[2] and Sofija Bogdanova[3]

Abstract **- Many watermarking techniques for image watermarking were proposed in the last few years. At the same time the large numbers of attacks are presented. Watermarking algorithm developers need a tool, which may help them to test the robustness of their proposed algorithms to these attacks. The tool that implements most of the standard image processing operations (watermarking attacks) is presented in this paper. The main characteristics of the proposed tool are: GUI - Graphical User Interface, simplicity of use and speed.**

Keywords **– Watermarking, Attacks, Evaluation, GUI.**

I. INTRODUCTION

We are witnesses of enormous growth of Internet and Internet sharing applications and data. The advancement in technology offers new solutions, but in the other way, creates new problems as well. Digital media (e.g. audio, images, video, etc.) offers many benefits: It can be stored, duplicated and distributed everywhere in the world, with no loss of fidelity, but in contrary, it can also be manipulated and modified easily, often only with personal computer and appropriate software, and sometimes even unintentionally. While these properties are desirable in general, they can create problems for parties who own digital media and want to distribute it, but at the same time, want to protect it from illegal multiplication and distribution.

Because of that, there is a need for protecting the intellectual property rights. Digital watermarking has been proposed as a solution for the copyright protection. It is a process of embedding hidden copyright information directly into the digital data by making small, unnoticed for human eye, modifications to them.

A digital watermark should possess certain properties, although their relative importance can vary depending upon the application [1]. The most important characteristics for effective watermarking are invisibility and robustness. This means that the watermark should be embedded into the image so as to be invisible over all image types. Also, it should be robust to intentional or unintentional image processing operations, which preserve the desired image quality.

[1]Jovan Spasik is with the Faculty of Electrical Engineering, University "Ss. Cyril and Methodius", 1000 Skopje, R.Macedonia, E-mail: vanja@mail.net.mk

[2]Dimitar Taskovski is with the Faculty of Electrical Engineering, University "Ss. Cyril and Methodius", 1000 Skopje, R.Macedonia, E-mail: dtaskov@etf.ukum.edu.mk

[3]Sofija Bogdanova is with the Faculty of Electrical Engineering, University "Ss. Cyril and Methodius", 1000 Skopje, R.Macedonia, E-mail: sofija@etf.ukim.edu.mk

A lot of watermarking algorithms that meet these two main requirements for effective watermarking have been proposed in recent years. An overview of more than 100 proposed methods is given in [2].

In the period of time required for marked data to reach the watermark receiver, an embedded watermark may be accidentally, inadvertently, but most of the time purposely, impaired from some processes. Any processing that may impair detection of the watermark is called watermark attack. An attack is succeeds if it impairs the watermark beyond acceptable limits while maintaining the perceptual quality of the attacked data. All attacks could be performed intentionally or unintentionally. Unintentional attacks are results of common signal processing operations done by legal user of watermarked works. Intentional attacks are usually performed by more competent people with more knowledge of watermarking systems and more resources to make the attacks.

As fast as new watermarking algorithms are proposed the large numbers of attacks are presented [3-6]. These attacks have shown that far more research is required to improve the quality of existing watermarking methods.

The complete theoretical analysis of the watermarking algorithm performance with respect to different attacks is rather complicated. In general, the developers of watermarking algorithms refer to the results of experimental testing, and they usually claim that proposed watermarking method is robust although only few experimental tests (e.g. JPEG compression, insertion of noise, cropping, etc) are performed to only few images. In order to overcome this problem with evaluation of the watermarking algorithm performance several benchmarks are proposed [7-9]. The benchmarks usually combine some of the possible attacks into a common framework and weight the resulted performances depending on the possible application of the watermarking technology. All these benchmarks include tools for watermarking attacks implementation. The main disadvantages of this tools are that some of them are very complex for use, some of them do not offer GUI, parameters of some predefined attack cannot be change and they do not include some of the standard attacks.

In this paper we present a GUI based tool that implement most of the standard image processing operations. This tool can help watermarking algorithm developers to test the robustness of their proposed algorithms. Benefit of this proposed tool, beside GUI, is that in same place it comprises many standard operations for image processing - watermark attacks, saves quality time, it may work on multiple images simultaneously and it is very simple for use.

In the next section, we briefly summarize the watermarking attacks and coarsely categorize them. Then, we propose GUI based tool that implements most of these attacks, classified in two main groups: simple and geometrical attacks.

II. CLASSIFICATION OF WATERMARKING ATTACKS

The watermark attacks can be classified in many different ways. In this paper we followed the categorization proposed in [4] where watermarking attacks are grouped in four main groups: Simple attacks, Geometrical attacks, Ambiguity attacks and Removal attacks.

A. Simple attacks

The main characteristic of this type of watermark attacks is that they manipulate on whole image and therefore attacks the embedded watermark. So they do not try to separate or to identify the watermark from the host image. Examples may include linear and general nonlinear filtering, JPEG compression, addition of noise and gamma correction.

B. Geometrical attacks

Sometimes the affect of the attack over the watermarked image is to make the image undetectable by watermark detector. These types of attacks disable the synchronization of the watermark detector, and usually include geometric transformation like cropping, rotation, shifting, permutations or removal of pixels or any other geometric transformation of the data. The main goal of these attacks is to make the watermark unreadable even though it is still present in the modified image.

C. Ambiguity attacks

Attacks often can be made and by insertion of confusion with creation of fake original or watermarked data. In other cases this attacks can discredit the authority of original watermark by embedding one or several additional watermarks such that it is unclear which the first, original watermark was.

D. Removal attacks

These are the most sophisticated attacks since they take into account prior knowledge about watermarking process. They first analyze the image, and than try to separate the watermark from the host data. After the removing of watermark, the original image is vulnerable to further illegal use. Examples are collusion attacks, denoising and some non-linear filter operations.

It should be noted that the properties of the attacks make its classification very hard, and then an attack may belong to more than one group. Cropping for example can be regarded as either a simple attack or a geometrical attack.

III. GUI BASED TOOL FOR WATERMARKING ATTACKS

We have chosen MATLAB [10] as the environment for developing our GUI. MATLAB contains extensive library of built-in functions that greatly simplifies programming and most of the watermarking system designers first implement their algorithms in MATLAB. They also need a tool for testing the performances of their algorithms.

With the proposed tool (Fig. 1) the process of testing is simplified and unified. This tool also saves time, because in one turn more than one image and attack can be chosen. The name of the modified images is compound from the names of the used attacks, and just one look at the name of the images shows which attack is implemented on that image.

In this tool the available attacks are classified in 2 groups: Geometrical and Simple attacks. The realization of other two groups of attacks will be discussed in further releases.

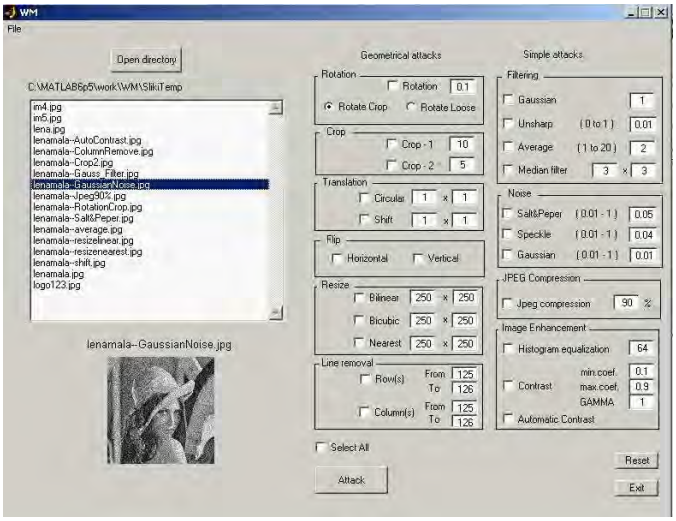

Fig.1. GUI based tool for watermarking attacks

We do not make difference between intentional and unintentional attacks.

The components of *Geometrical attacks* group are:

Rotation: This attack does not usually change the commercial value of the image but can make the watermark un-detectable. Rotations are used to realign horizontal features of an image. In this tool rotate attack can be made in two ways (with or without crop).

Cropping: Usually, attackers are just interested in the central part of the copyrighted material, so in this case cropping is ideal solution for breaking the synchronization and making recovery of the watermark impossible.

Translation: Disabling the synchronization can also be easily done with this type of attack. Proposed tool presents two translation attacks: shifting in one direction and circular shift

Flip: The very few watermarking systems can survive horizontal and vertical flip, although there is no loosing of information in the image with this type of attacks.

Resize: Resizing of the image can be done using three different interpolation methods: bilinear, bicubic and nearest. A lot digital watermarking methods are resistant to this kind of attacks.

Line removal: This attack can removes selected number of row(s) and column(s) from the image and it is very efficient against any simple implementation of spread-spectrum techniques in the spatial domain.

The second group named *Simple attacks* is compound of four subgroups: Filtering, Noise, JpegCompression and Image Enhancement group.

Commonly used linear and non-linear filters are: symmetric Gaussian low pass filter, standard average filter and median filter. In Filtering group there is also sharpening filter. These filters can be used as an effective attack on some watermarking schemes.

In the communication theory and signal processing theory literature, additive noise and uncorrelated multiplicative noise have been largely addressed. Authors often claim that their watermarking techniques survive this kind of attack, however many forget to mention the maximum level of acceptable noise. Proposed tool includes these types of noises:

Salt & pepper - adds "salt and pepper" noise to the image,

Speckle - adds multiplicative noise to the image,

Gaussian - adds Gaussian white noise to the image.

One of the most widely used intentional or unintentional attacks is JPEG compression algorithm. Because of that, every watermarking system should be flexible to some degree of JPEG compression. In this tool we can experiment with different values of compression by changing quality factor in the edit box. Here we can point out that attackers often use this attack in combination with some geometrical transformation (e.g. cropping and rotation).

The final group of attacks is named Image Enhancement. These attacks usually do not prevent watermark detection, and they are often applied before detection is performed to obtain better results. Histogram equalization is attack that includes histogram stretching or equalization, which are sometimes used to compensate poor lightening conditions. Contrast attack adjusts image intensity values. Here we can choose Gamma factor that specifies the shape of the curve describing the relationship between the values in input and output picture.

IV. CONCLUSIONS

The aim of this tool is to help the watermark algorithm developers to test the robustness of their watermarking algorithms. This can be done by testing the behavior of watermarked image, attacked by most of the existing attacks. In proposed tool this could be done from the same location in just one turn. Beside this, the presented tool is easy for using thanks to GUI, saves quality time and may work on multiple images simultaneously.

REFERENCES

[1] I. Cox, M. Miller and J. Bloom, “Watermarking applications and their properties,” Int. Conf. on Information Technology’2000, Las Vegas, 2000.

[2] F. Hartung and M. Kutter, “Multimedia watermarking techniques,” Proceedingg of the IEEE, Special Issue on Protection of Multimedia Content, vol. 87, July 1999.

[3] F.A.P. Petitcolas, R.J. Anderson and M.G. Kuhn, “Attacks on copyright marking systems” in Second Workshop on Information Hiding, Portland, OR, USA, Apr. 1998.

[4] F. Hartung, J.K. Su, and B. Girod, “Spread Spectrum Watermarking: Malicious Attacks and Counterattacks”, Proc. of SPIE: Security and Watermarking of Multimedia Contents, vol. 3657, San Jose, CA, Jan 1999.

[5] S. Voloshhynovskiy, S. Pereira, A. Herrigel, N. Baumgartner and T. Pun, “Generalized watermark attack based on watermark estimation and perceptual remodulation,” in Proc. Electronic Imaging 2000, Security and Watermarking of Multimedia Contents II, vol. 3971, San Jose, CA, Jan 2000.

[6] M. Kutter, “Watermark copy attack,” in Proc. Electronic Imaging 2000, Security and Watermarking of Multimedia Contents II, vol. 3971, San Jose, CA, Jan. 2000.

[7] http://www.cl.cam.ac.uk/~fapp2/watermarking/stirmark

[8] V. Solachidis, A. Tefas, N. Nikolaidis, S. Tsekeridou, A. Nikolaidis, I.Pitas, “A benchmarking protocol for watermarking methods”, 2001 IEEE Int. Conf. on Image Processing (ICIP'01), Thessaloniki, Greece, October, 2001

[9] Shelby Pereira, Sviatoslav Voloshynovskiy, Maribel Madueño, Stéphane Marchand-Maillet and Thierry Pun, “Second generation benchmarking and application oriented evaluation,” in Information Hiding Workshop III, Pittsburgh, PA, USA, April 2001.

[10] http://www.mathworks.com